

**CLAIMS**

Now, therefore, the following is claimed:

- 1           1.       A computer system for processing instructions of a computer program,  
2       comprising:  
3           a plurality of registers;  
4           a plurality of connections corresponding respectively with said registers;  
5           a pipeline configured to process and execute at least one of said instructions,  
6       said one instruction associated with a register identifier that identifies one of said  
7       registers;  
8           a scoreboard coupled to said plurality of connections and to said pipeline, said  
9       scoreboard having a plurality of bits corresponding respectively with said plurality of  
10       registers, said scoreboard configured to receive said first register identifier and, in  
11       response to said first register identifier, to assert one of said bits that corresponds with  
12       said one register, said scoreboard configured to receive a second register identifier  
13       identifying at least said one register and, in response to said second register identifier,  
14       to deassert said one bit, said scoreboard further configured to transmit said one bit  
15       across one of said connections, said one bit indicative of whether a pending write to  
16       said one register exists based on a value of said one bit and based on which of said  
17       connections is transmitting said one bit; and  
18       hazard detection circuitry coupled to said plurality of connections, said hazard  
19       detection circuitry configured to receive said one bit and to detect whether a data  
20       hazard exists based on said one bit.

1           2.       The system of claim 1, wherein said scoreboard is further configured to  
2       transmit a data word including each of said plurality of bits to said hazard detection  
3       circuitry, each asserted bit in said data word indicating that a different one of said  
4       registers is associated with a pending write.

1           3.       The system of claim 1, wherein said scoreboard includes a plurality of  
2       registers, each of said registers containing a different one of said bits and connected to  
3       a different one of said connections.

1           4.       The system of claim 1, wherein said first register identifier is a decoded  
2       register identifier, said system further comprising:  
3               a decoder configured to receive an encoded register identifier defined by said  
4       instruction and to decode said encoded register identifier into said decoded register  
5       identifier, said decoder further configured to transmit said decoded register identifier  
6       to said scoreboard.

1           5.       The system of claim 4, wherein said decoder is further configured to  
2       transmit said decoded register identifier to said hazard detection circuitry.

1           6.       The system of claim 4, wherein said decoded register identifier  
2       includes a plurality of bits, wherein one of said bits in said decoded register identifier  
3       that corresponds to said one register is asserted, and wherein the remaining bits in said  
4       decoded register identifier are deasserted.

1           7.       A system for processing instructions of computer programs,  
2 comprising:  
3           a plurality of registers;  
4           a plurality of connections, each of said connections corresponding to a  
5 different one of said registers;  
6           means for maintaining a plurality of bits and for indicating via said bits which  
7 of said registers is associated with a pending write, said maintaining means configured  
8 to transmit said bits across said connections, wherein each bit transmitted across each  
9 of said connections is indicative of whether the register corresponding to said each  
10 connection is associated with a pending write; and  
11          hazard detection circuitry configured to detect data hazards based on said bits  
12 transmitted across said connections.

1           8.       The system of claim 7, wherein said maintaining means includes a  
2 plurality of registers, each of said registers containing a different one of said bits and  
3 connected to a different one of said connections.

1           9.       The system of claim 7, further comprising:  
2           means for processing an instruction; and  
3           means for decoding an encoded register identifier associated with said  
4 instruction into a decoded register identifier, said decoding means configured to  
5 transmit said decoded register identifier to said hazard detection circuitry and to said  
6 maintaining means, said decoded register identifier identifying one of said registers,  
7           wherein said hazard detection circuitry is configured to detect a data hazard  
8 based on said decoded register identifier and said maintaining means is configured to  
9 modify one of said bits based on said decoded register identifier, said one bit  
10 corresponding to said one register.

-1           10.     The system of claim 9, wherein said decoded register identifier  
2 includes a plurality of bits, wherein one of said bits in said decoded register identifier  
3 that corresponds to said one register is asserted, and wherein the remaining of said bits  
4 in said decoded register identifier are deasserted.

1           11.     A method for processing instructions of computer programs,  
2     comprising the steps of:  
3           providing a plurality of registers;  
4           maintaining a plurality of bits, each of said bits indicating whether a  
5     corresponding one of said registers is associated with a pending write;  
6           transmitting a data word, said data word including each of said bits, wherein  
7     each asserted bit in said data word indicates that a different one of said registers is  
8     associated with a pending write;  
9           receiving said data word; and  
10          detecting a data hazard based on said data word.

1           12.     The method of claim 11, further comprising the steps of:  
2           processing an instruction;  
3           receiving an encoded register identifier associated with said instruction;  
4           decoding said encoded register identifier into a decoded register identifier;  
5           comparing said decoded register identifier to another register identifier;  
6           detecting a data hazard based on said comparing step; and  
7           modifying one of said bits based on said decoded register identifier.

1           13.     A method for processing instructions of computer programs,  
2     comprising the steps of:  
3           providing a plurality of registers;  
4           maintaining a plurality of bits, each of said bits respectively corresponding  
5     with one of said registers;  
6           providing a plurality of connections, each of said connections respectively  
7     corresponding with one of said registers;  
8           indicating, via said bits, which of said registers are associated with pending  
9     writes;  
10          transmitting one of said bits corresponding with a particular one of said  
11     registers across a particular one of said connections, said particular one connection  
12     corresponding with said particular one register; and  
13          detecting a data hazard based on said one bit transmitted across said particular  
14     one connection.

1           14.     The method of claim 13, further comprising the steps of:  
2           processing an instruction;  
3           receiving a register identifier associated with said instruction;  
4           comparing a bit of said register identifier with said particular one bit  
5     transmitted across said particular one connection; and  
6           performing said detecting step based on said comparing step.

- 1        15.     The method of claim 13, further comprising the steps of:
- 2        processing an instruction;
- 3        receiving an encoded register identifier associated with said instruction;
- 4        decoding said encoded register identifier into a decoded register identifier;
- 5        comparing said decoded register identifier to another register identifier;
- 6        detecting a data hazard based on said comparing step; and
- 7        asserting one of said bits based on said decoded register identifier.